$B_2 = 0,6 \times 4 + 0,4 \times 3 = 3,6$ – лидер,
$B_7 = 0,6 \times 3 + 0,4 \times 2 = 2,6$.
Информационный комитет:
$B_4 = 0,6 \times 5 + 0,4 \times 6 = 5,4$ – лидер,
$B_8 = 0,6 \times 4 + 0,4 \times 1 = 2,8$,
$B_9 = 0,6 \times 3 + 0,4 \times 0 = 1,8$,
$B_{10} = 0,6 \times 3 + 0,4 \times 2 = 2,6$.

6. Выводы

Как показывает практика общественных организаций и органов студенческого самоуправления, все более сложным и более актуальным является именно грамотное распределение членов организации в команды (комитеты).

В статье рассмотрена модель определения глав для трех комитетов, которые функционируют в работе студенческого сената ХНУРЭ.

Модель основана на социометрических методах, которые принадлежат к поведенческому подходу менеджмента и строятся на пожеланиях участников коллектива по отношению к составу групп.

Таким образом, процесс формирования комитетов и определение глав для них является очень важной проблемой для общественных организаций. Поэтому применение этой модели на практике позволяет избежать ошибок в работе общественной организации.

Литература

1. Вітлінський В. В. Моделювання економіки [Текст] / : Навч. посібник. – К.: КНЕУ, 2003. – 408 с.
2. Большаков А. С. Менеджмент [Текст] / : Учебное пособие. — СПб.: «Издательство "Питер"», 2000. — 160 с.
3. Орлов А.И. Менеджмент [Текст]: учебник. – М.: Издательство «Изумруд», 2003. — 298 с.
4. Статут Студентського сенату Харківського національного університету радіоелектроніки від 19.10.2010.
5. Івченко Н.Б., Математичні моделі та методи в менеджменті, маркетингу й економіці [Текст] / : Навч. посібник. – Х.: Компанія СМІТ, 2007. – 168 с.

УДК 004.932

RECOGNITION OF MATHEMATICAL NOTATION

T. Iakovenko
Faculty of Computer Sciences
Kharkiv National University of Radioelectronics
Lenina str., 14, Kharkiv, Ukraine, 61116
Contact phone: 093-466-94-11
E-mail: tanya.yakovenko3@gmail.com

Розглянуті основні методи, які використовуються для розпізнавання математичних формул в електронних документах. Зроблена порівняльна характеристика різних програмних реалізацій, як INFTY, FineReader, CuneiForm

Ключові слова: OCR, физична та логічна сегментація, LaTeX

Рассмотрены основные методы, которые используются для распознавания математических формул в электронных документах. Проведена сравнительная характеристика различных программных реализаций, как INFTY, FineReader, CuneiForm

Ключевые слова: OCR, физическая и логическая сегментация, LaTeX

The article represents main methods, which uses for recognition of mathematical formulas in the electronic documents and comparative analysis for the different software implementations, as INFTY, FineReader, CuneiForm

Key words: OCR, physical and logical segmentation, LaTeX

1. Introduction

Over the past few years there has been a general trend of electronic documents in favor of digital and electronic storage options. As it becomes easier for the user to create electronic documents and publish information through the Internet. Many authors and researchers are taking advantage of it, they publish their articles and work even earlier than they appear in magazines and books. There is a growing number of scientific papers, articles and publications available in electronic and digital form. As a consequence, there is a need of organizing, recording and understanding these documents. The biggest complexity is the recognition of mathematical notation, because nowadays we don't have

an ideal algorithm which has 100 percent accuracy and does not need human support for every recognition. Extracting mathematical formulas is qualitatively different from the extraction of plain text because it is a much more complex task. It involves search and segmentation of regions containing mathematical formulas in the body document and identification of individual elements of the formulas and save the results in a convenient form.

2. Recognition of mathematical notation

The first step of OCR for every document is to create 3 streams, one of the steams will include only text, the second stream will have mathematical symbols, and the third will have diagrams, tables, logos and pictures. This will help get higher accuracy for mathematical recognition, because we can use a special recognizer that focuses only on the mathematical symbols. We have two types of mathematical symbols which can be found in the documents. They are called display and in-line mathematics. Display mathematics are easily to recognize, because they have more spaces around every formula. They could be detected as non-text, low density, unusual line statistic and smaller words. Also they have a special regions on the page. In-line mathematical symbols is a much harder case. A lot of magazines try to save space, to put as much in-line symbols as they can. Here is just one example : $A=\pi r$, as you see it is the most indiscernible case, unless you have some context for formulas. It is a big problem for segmentation [1].

We have the possibility to improve our accuracy among in-line math symbols. If documents have TeX source, recognition of mathematical symbols would be much easier. This is because all mathematical symbols will have a math-mode or text within. And when we divide these symbols as mathematical and text, we can put all possible symbols to the math-box. If something was not recognized by the special engine, a user can figure it out. After these procedures we can get less errors, because the new symbols could be identified. These symbols can then be changed in the text-box and then re-group the math symbols which were mistakenly grouped during the automatic process.

To avoid some errors, when you recognize the math formulas were recognized as text, you can use these rules:

- A left parenthesis without math to the right.
- A trailing right parenthesis whose matching left parenthesis is missing from this clump.
- A horizontal line unless it has math to its right, above or below.
- A leading or trailing comma or dot in the clump.
- An isolated 1 or even 11 in the math bag that it is within a single-character-space of text, which may instead be l or an ll. (We treat 1 and l equivalently).
- An isolated 0 or even 00 in the math bag that it is within a single-character-space of text, which may instead be o or an oo. (We treated 0 and 0 equivalently).

Mathematical symbols have different size, location of text lines, the letter and symbol frequencies, unusual position, and layouts. After OCR it should not go directly to ASCII text, before we have to have some meta–level language.

Nowadays we do not have any special OCR solution which can help recognize in all mathematic symbols. The OCR system will try to decode all mathematical symbols as text, pushed to baseline and in the end we can get only mess. The other solution is to get the image of the formula which will give higher accuracy and the possibility to manipulate and evaluate it.

3. Segmentation methods of mathematical notation

Lee and Wang techniques:

Idea of the method is all mathematical symbols based on the different criteria. Examples are higher symbols and difference in spacing between symbols. These are a good criteria for the first math recognition, but it should have more steps. Because it gives a lot of mistakes one example is that headers are recognized as a formula and the method is not capable to provide a formula with similar symbols. Characters that are known as math symbols are used to add to the roots of the sub trees, adjacent characters are added to the tree level below.

Methods of physical and logical segmentation:

The method consists of two stages: The physical stage and the logical segmentation stage. At the stage of the physical segmentation the body of the document is divided into text boxes, lines, words and symbols. Then, in the logical segmentation stage it allocates only formulas. For the selection of isolated formulas it uses similar assumptions as the Lee method. The set of the special characters finds the in-line formula. For the display mathematics it uses the following rules:

- «+», «=», «<» or ">"- captured the characters around;
- brackets - everything between the brackets;
- fraction bar - all above and below the line [2].

Fateman's method:

Fateman's method uses 3 stages for the recognition of mathematic notation. First stage is pre-initialized – we have 2 empty "baskets" which should be complete by mathematics and text. During the first stage symbols are added to the mathematical "basket" according to the rules:

- special mathematical symbols: +, =, Greek letters, scientific symbols, large brackets, horizontal lines;
- text in italics and bold;
- figures;
- lower and upper indices;
- parentheses (), [];
- points, commas, colons, semicolons and other punctuation.

Everything else is regarded as text and placed into the second "basket". As a result, the "basket" with the math accumulated surplus set of characters of which some can not apply the mathematic notation (e.g. punctuation).

In the second stage, based on elements in the mathematical baskets we construct the zone of proximity - a group of characters, which includes components which are adjacent to each other.

We obtain a domain, consisting supposedly of mathematical elements, that is a free-standing mathematical expression. Single mathematical symbols or "basket" is sufficiently far from other mathematical symbols, and, simultaneously, located near the symbols of text "baskets" are considered as text and moved into the text "basket". The criteria of proximity is used for the distance between the minimum bounding rectangle (bounding box) of each character. The characters are considered adjacent if the distance between them is less than some threshold.

At the end of the group, the areas proximity correction is performed (it removes leading and trailing periods and commas, the extra brackets, etc.). In the third stage mathematical words are allocated such as sin, cos, log.

As a result, these groups of elements are segmented from the document formula.

In general, this approach gives acceptable results, but it still can have systematic errors:

• Italic's row are considered as a mathematician

• Some expressions, such as "(2)", "1:" also are regarded as a mathematical notation.

Nowadays there is software that can find text recognition, as well as recognize mathematic symbols [3].

Cuinie and FineReader use the same methods for symbol recognition. At first, it divides pages on different groups as text and pictures. After it converts text fragments to text it compares every symbol with a raster pattern. Symbol's are superimposed with patterns in the database and then it chooses the pattern with the minimum points differing from original symbol. When the document has a really bad quality, the structural recognition method of a character's distorted image stands out. It then compares the specific details and compares within the structural patterns of the symbols. As a result, the character is choosing with the set of all structural elements and their location is the best matching to the recognizable symbol.

One of the most successful software for the mathematical recognition is INFTY, which was developed by Japanese scientists. They use two complementary recognition engines, a commercial OCR engine for ordinary texts and an original recognition engine for mathematical expressions. The main stages for recognition are: layout analysis, character recognition, structure analysis of mathematical expressions and manual error correction [1].

In the layout analysis procedure, which is the first procedure of INFTY, several preprocessing operations, such as binarization, noise removal, and deskewing, are performed on the page images (scanned in 600dpi) of a mathematical document. After all connected components are extracted from the preprocessed page image, they are separated into figure / table areas and non-figure areas. The non-figure area is further decomposed into text lines.

The character recognition procedure separates each text line into mathematical expressions and ordinary texts and recognize the characters on both ordinary texts and mathematical expressions.

Structure analysis of mathematical expressions has three roles in INFTY. The first role is to represent the structure of each mathematical expression by a tree for converting the mathematical document into XML, LaTeX, and other math-description formats. The second role is to fix the character recognition result of the mathematical expressions. The third role is to detect the ordinary texts wrongly classified into the mathematical expression part.

The structure analysis problem is can now be considered as the problem of searching for the minimum-cost spanning tree on the weighted digraph prepared in the previous section. The spanning-tree is somewhat peculiar and different from common spanning-trees which will consist of all nodes of the digraph and it consists of only nodes which do not correspond to the same connected component.

Once the minimum-cost spanning-tree of the weighted digraph is obtained, the structure of the subjected mathematical expression is represented as a (spanning-) tree and the character recognition result is fixed. Thus, the first and the second role noted at the beginning of this section are fulfilled by this procedure. On the other hand, if the spanning-tree is not obtained, the mathematical expression is rejected as an ordinary text. Thus, the third role can be fulfilled along with this procedure. Manual error can be corrected using a graphical user interface.

From results on 476 pages of mathematical documents it was shown recognition rates (99.44% on ordinary texts, 95.18% on mathematical expressions, and 98.51% in total) and 89.6% mathematical expressions are perfectly analyzed [4].

4. Conclusions

Mathematics recognition is a practically-important and hard problem. Recognition is difficult because mathematical notation involves a large alphabet of symbols and a large range of font sizes, and contains little redundancy in its representation of information. Comparison among existing mathematics-recognition systems is complicated by the myriad ways in which the mathematics-recognition problem can be defined. In addition, some recognizers assume pixels as input, whereas others assume that symbol-recognition has already taken place. This great variation in problem-definition makes it difficult to compare the strengths and weaknesses of existing mathematics recognition systems. One of the best software implementation is INFTY, which gives the best results of formula's recognition and Fateman's method recognize more symbols then Lee and Wand techniques or logical and physical segmentation.

Literature

1. Recognition of Mathematical Notation - Dorothea Blostein, Ann Grbavec. - Handbook on Optical Character Recognition and Document Image Analysis, Eds. P.S.P. Wang and H. Bunke, Chapter 22.
2. http://www.ocf.berkeley.edu/~mlyang/papers/MichaelYangPsmath.pdf.
3. http://www.fi.muni.cz/usr/sojka/download/dml2008/14.pdf.
4. Infty-an integrated OCR system for mathematical documents - M.Suzuki, F.Tamari, R.Fukuda, S.Uchida, T.Kanahori. - Proceedings of ACM Symposium on Document Engineering 2003, Grenoble, Ed.C.Vanoirbeek, C.Roisin, E. Munson, 2003, pp.95-104.